(2)

NPSAW - 91 - 001

NAVAL POSTGRADUATE SCHOOL

Monterey, California

TWO RANDOM TOUR PROCESSES OF KNOWN LENGTH BETWEEN KNOWN END POINTS

by

R.N. FORREST

JUNE 1991

Prepared for:
Naval Postgraduate School
Monterey, CA 93943-5000

NAVAL POSTGRADUATE SCHOOL
Monterey, California

Rear Admiral R. W. West, Jr.
Superintendent

Harrison Shull
Provost

This report was prepared in conjunction with research conducted for the Chief of Naval Operations and funded by the Naval Postgraduate School:

R. N. Forrest
Professor of
Operations Research

Reviewed by:

James N. Eagle, Chairman
ASW Academic Group

Released by:

Paul J. Marto, Dean of
Research

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved OMB No 0704-0188

1a REPORT SECURITY CLASSIFICATION
UNCLASSIFIED

1b RESTRICTIVE MARKINGS

2a. SECURITY CLASSIFICATION AUTHORITY

2b DECLASSIFICATION / DOWNGRADING SCHEDULE

3. DISTRIBUTION / AVAILABILITY OF REPORT
Approved for public release; distribution is unlimited.

4. PERFORMING ORGANIZATION REPORT NUMBER(S)
NPSAW-91-001

5 MONITORING ORGANIZATION REPORT NUMBER(S)

6a NAME OF PERFORMING ORGANIZATION
Naval Postgraduate School

6b OFFICE SYMBOL (If applicable)
AW

7a NAME OF MONITORING ORGANIZATION
Chief of Naval Operations

6c. ADDRESS (City, State, and ZIP Code)
Monterey, CA 93943-5000

7b ADDRESS (City, State, and ZIP Code)
Washington, DC 20350

8a. NAME OF FUNDING / SPONSORING ORGANIZATION
Naval Postgraduate School

8b OFFICE SYMBOL (If applicable)

9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
O & MN, Direct Funding

8c. ADDRESS (City, State, and ZIP Code)
Monterey, CA 93943-5000

10 SOURCE OF FUNDING NUMBERS

PROGRAM ELEMENT NO	PROJECT NO	TASK NO	WORK UNIT ACCESSION NO

11 TITLE (Include Security Classification)
TWO RANDOM TOUR PROCESSES OF KNOWN LENGTH BETWEEN KNOWN END POINTS

12 PERSONAL AUTHOR(S)
R. N. Forrest

13a TYPE OF REPORT	13b TIME COVERED	14 DATE OF REPORT (Year, Month, Day)	15 PAGE COUNT
Technical	FROM ______ TO ______	1991 June	30

16 SUPPLEMENTARY NOTATION

17 COSATI CODES

FIELD	GROUP	SUB-GROUP

18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
Random Tracks
Track Generating Programs

19 ABSTRACT (Continue on reverse if necessary and identify by block number)

The generation of random tracks of specified length between known points is the subject of the report. The tracks lie is a plane and consist of a sequence of connected legs of random orientation and length. A program is listed in the report that generates a graphical representation of a track as well as data in terms of leg course and leg length that is sufficient to define the track.

20 DISTRIBUTION / AVAILABILITY OF ABSTRACT
☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS

21 ABSTRACT SECURITY CLASSIFICATION
UNCLASSIFIED

22a NAME OF RESPONSIBLE INDIVIDUAL
R. N. Forrest

22b TELEPHONE (Include Area Code)
(408) 624-9098

22c OFFICE SYMBOL
AW

DD Form 1473, JUN 86 Previous editions are obsolete

S/N 0102-LF-014-6603

SECURITY CLASSIFICATION OF THIS PAGE

UNCLASSIFIED

The programs in this report are presented without representation or warranty of any kind.

Table of Contents

I. Introduction

This report is an extension of Reference 1. It is about tracks of specified length that are generated by objects that move in a plane between two points on a sequence of legs of random orientation and length. Figure 1 represents a map of the track of an object that moved in this manner.

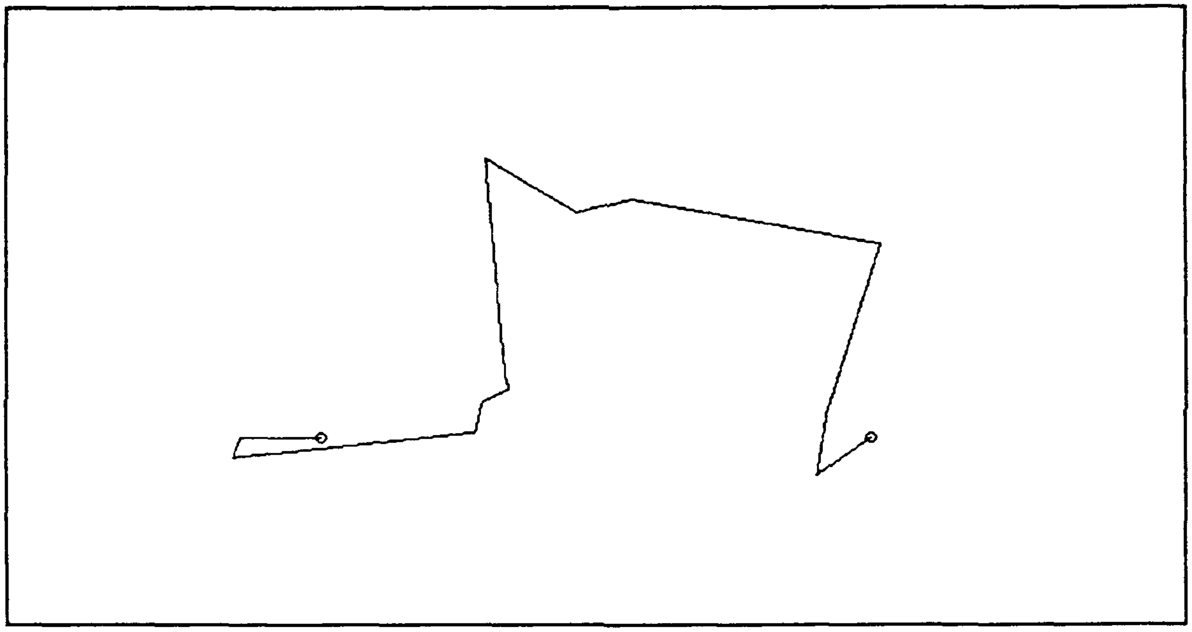

Figure 1. A line that represents a track of specified length that was generated by an object that moved in a plane between two points on a sequence of legs of random orientation and length.

In a formal sense, the tracks are realizations of two dimensional stochastic processes. The characteristics of these processes are described in more detail in Section II of this report and some mathematical relations associated with them are developed in Appendix 1.

The program that is described in Section III will define a process and simulate its realizations. The output of the program includes a graphical representation of a track, a table listing the course and length of each track leg and some of the track's statistical characteristics. The program which is written in BASIC is listed in Appendix 2. Figure 1 is an example of the graphical output of the program.

II. The Realizations of a Process

The realizations of a process are tracks that are lines in a plane of specified length between two specified points. The lines consist of a sequence of connected straight legs whose orientation and length are determined by the process.

A sequence of right handed rectangular coordinate systems can be used to define a track. The first is a reference system for the first leg, the second is a reference system for the second leg and so on to the final leg. For each coordinate system, the x-axis is coincident with a line joining the initial point of the leg and the end point of the track, the origin is at the midpoint of that line and the x-axis is oriented so that the x-coordinate of the end point of the track is positive.

A sequence of polar coordinate systems also can be used to define a track. The first is a reference system for the first leg, the second is a reference system for the second leg and so on to the last leg. For each coordinate system, the polar axis of the system is parallel to the positive y-axis of the leg's rectangular coordinate system, the origin is at the initial point of the leg and angular coordinates are positive when measured in the clockwise direction. With this convention, relative to the positive y-axis of the leg's rectangular coordinate system, the angular coordinate of the end point of the leg is its bearing from the initial point of the leg as well as the course of an object while moving on the leg.

In addition to the two coordinate systems, each leg is associated with an ellipse that is a boundary curve for the leg's end point. This bounding ellipse has one focus at the initial point of the leg and the other focus at the end point of the track. And the distance of any point on the ellipse from the focus at the initial point of the leg plus the distance of the point from the focus at the end point of the track is equal to the remaining length of the track. Figure 2 illustrates the coordinate systems associated with the last two legs of a track. In this case, the second leg ends and the third leg begins on the second leg's bounding ellipse.

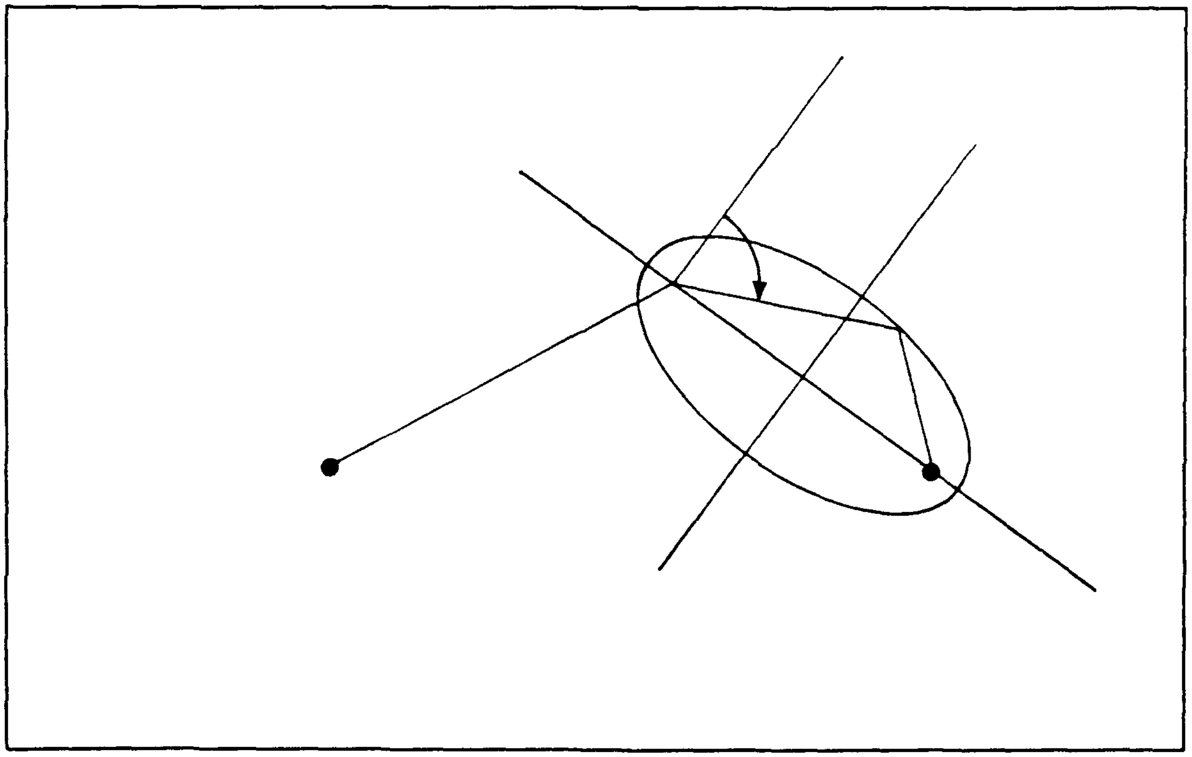

Figure 2. The coordinate systems and the bounding ellipse for the second leg of a three leg track. The ellipse major axis lies on the x-axis.

Two classes of processes are considered here. Those in the first class are referred to as *static processes* and those in the second class are referred to as *dynamic processes*. For either class, the coordinates of a leg's end point in its polar coordinate system are random variables associated with a process. In the following, for the k^{th} leg, they are represented by Θ_k and R_k.

For a dynamic process, for every leg except the last, Θ_k is a random variable determined by a distribution with density function:

$$f_{\Theta_k}(\theta_k) = \left(\frac{1}{2\pi}\right)\frac{b_k}{a_k - c_k \sin(\theta_k)}$$

where $-\pi/2 \leq \theta_k < 3\pi/2$, $2a_k$ is the remaining track length (the bounding ellipse major axis), $2c_k$ is the distance between the leg's initial point and the track's end point (the bounding ellipse foci), and $2b_k = 2(a_k^2 - c_k^2)^{1/2}$ (the bounding ellipse minor axis).

For each leg except the last, given $\Theta_k = \theta_k$, its radial coordinate R_k is equal to either $r_k = b_k^2/(a_k - c_k \sin\theta_k)$, the radial coordinate of a point on the leg's bounding ellipse, or to the value of a random variable R_k^* which is determined by a conditional gamma distribution with density function:

$$f_{R_k^*|\Theta_k}(r_k^*|\theta_k) = \frac{1}{\Gamma(n)\,[\delta(\theta_k)/n]^n}(r_k^*)^{n-1} e^{-\frac{r_k^*}{\delta(\theta_k)/n}}$$

where $0 \leq r_k^*$, n is a positive integer so $\Gamma(n) = (n-1)!$, the parameter $\delta(\theta_k) = (a_k - c_k)\,\delta/(a_k - c_k \sin\theta_k)$ is the expected value of R_k^* given an angular coordinate θ_k, and δ is its expected value given the angular coordinate $\pi/2$. If $R_k^* < r_k$, then the radial coordinate $R_k = r_k^*$ and $2a_{k+1} = 2a_k - r_k^*$ is the remaining track length. Otherwise, $R_k = r_k$ and the next leg is the last leg with length $2a_{k+1} = 2a_k - r_k$.

A static process differs from a dynamic process only in that a, b and c replace a_k, b_k and c_k in the definition of the density function for both θ_k and R_k^*.

The basis for the choice of the distribution of the random variable θ_k that determines the relative course of an object on a leg and the conditional distribution of the random variable R_k that determines the length of the leg is discussed next. The discussion is in terms of a dynamic process, however, the extension to a static process is immediate.

For a dynamic process, for every leg except the last, as a function of θ_k, the density function for θ_k and the expected value of R_k^* are both a maximum at $\pi/2$ and a minimum at $-\pi/2$ and $3\pi/2$. In addition, the distribution of θ_k is symmetric about $\pi/2$. This results in a leg's course being focused toward the track's end point and the expected value of its length being proportional to the focusing since $(a_k - c_k)/(a_k - c_k \sin \theta_k)$ is the ratio of the value of r_k for a point on the k^{th} bounding ellipse with angular coordinate θ_k to the value of r_k for the point with angular coordinate $\pi/2$. Note, $r_k = b_k^2/(a_k - c_k \sin \theta_k)$.

For a static process, the first leg determines this focusing effect for each of the remaining legs except the last.

If the initial point and end point of a static process are coincident and the distribution index $n = 1$ so that R_k^* has an exponential distribution, as the size of the first bounding ellipse (circle) is increased, the process approaches a random tour process of the kind defined by Washburn in Reference 2. If the initial point and end point of a static process are not coincident, as the size of the first bounding ellipse is increased, the process approaches a generalization of these random tour processes.

The choice of a gamma distribution to determine values for R_k^* was made in order allow one to reduce the number of legs with lengths significantly less than the average leg length on a track when this is appropriate.

III. A Track Simulation Program

A program that simulates realizations of the random track processes is listed in Appendix 2. The program is written in Microsoft Quick BASIC 4.5 and is for a PC with a VGA or MCGA capable monitor. This requirement is determined on Line 340 by the statement SCREEN 11. To change the requirement, change this statement. For example, SCREEN 2 supports CGA.

The program can be used to print track data files or generate tracks representing either realizations of a dynamic process or a static process. After the choice of a process is made, five inputs are required in order to use the program: (1) the distance between the track initial point and end point; (2) the track length; (3) delta, the maximum expected leg length; (4) the leg distribution index and (5) the maximum number of legs allowed in a track.

The pseudorandom numbers that are used to produce the tracks can be generated by either the BASIC random number generator or by an auxiliary random number generator. If the BASIC random number generator is chosen, a random number seed can be specified. After the required inputs to the program have been made, a track is produced that is displayed on a graphics capable monitor. Figure 1 and Figure 3 illustrate this output. The program user is then given the option of generating a replacement track. When this option is declined, the following options are provided for the last displayed track: (1) display the track data, (2) create a track data file or (3) print the track data as is illustrated by Table I.

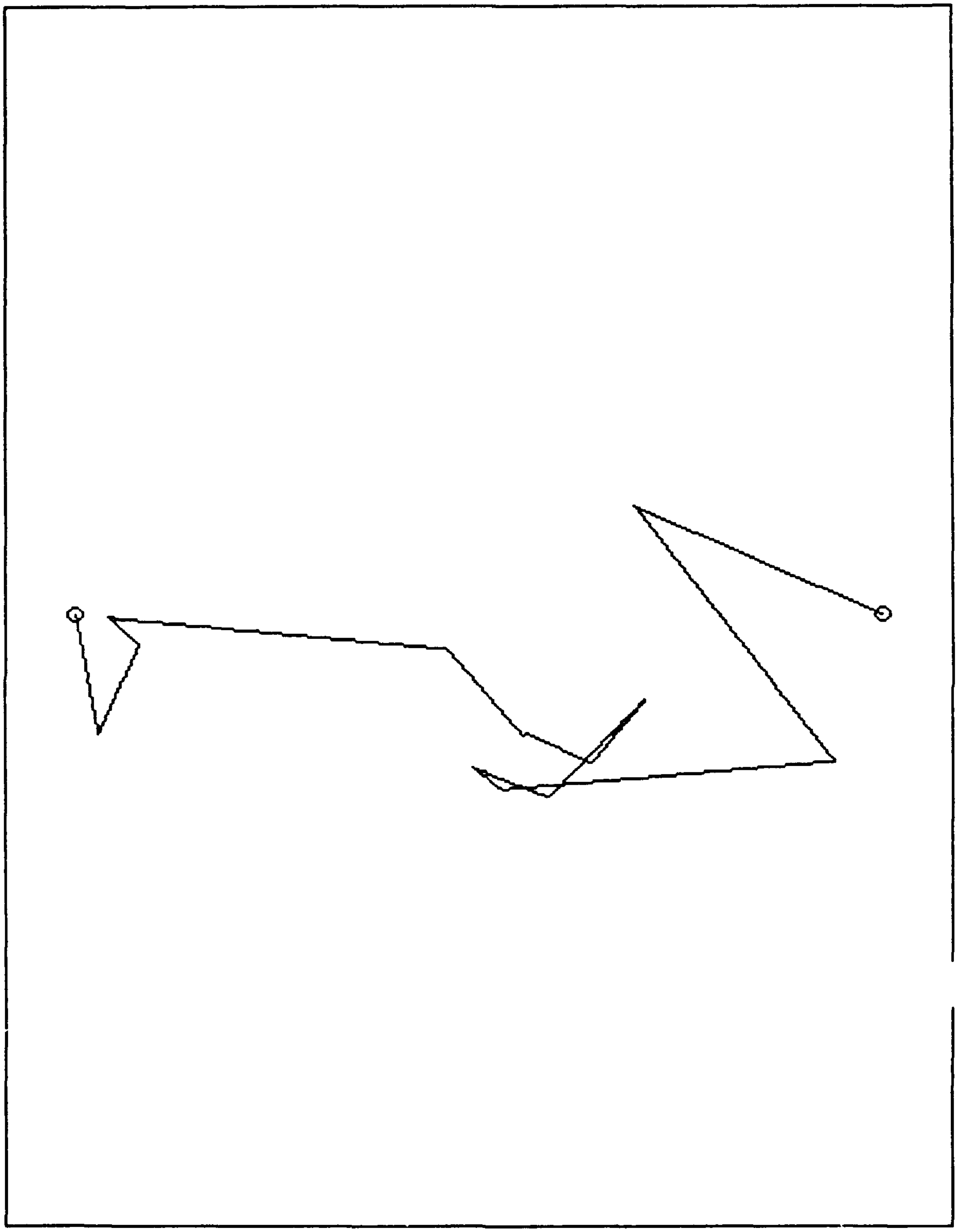

Figure 3. The track represents a realization of a static process. The track data is listed in Table I.

Table I Track data for the track shown in Figure 2. In the table, the first terminal leg is leg 13 and the second terminal leg is leg 14.

stationary leg bearing and range distributions
distance between the end points = 10
track length = 30
delta = 3
leg length distribution index = 1
maximum number of legs allowed = 30
number of legs = 14
standard random number generator
random number seed = 123

i	phi	leg length	theta	X	Y
0	0	0	0	-5	0
1	173	2.20348	173	-4.721061	-2.185753
2	17	1.749907	29	-4.21651	-.5101627
3	325	.6224251	328	-4.573226	-9.739542E-05
4	98	4.206195	98	-.411372	-.6092337
5	149	1.870394	156	.545191	-2.216518
6	82	6.720272E-02	109	.6117898	-2.207529
7	125	.9515256	152	1.389839	-2.755288
8	30	1.353381	67	2.064352	-1.581971
9	214	2.136891	243	.8589396	-3.346421
10	300	1.085471	339	-7.937231E-02	-2.800693
11	140	.5927657	169	.300589	-3.255666
12	83	4.141717	117	4.407269	-2.718077
13	332	5.341702	50	1.920847	2.009659
14	123	3.676944	90	5	0

track leg statistics for the nonterminal legs 1 through 12

maximum nonterminal leg length = 4.206195
average nonterminal leg length = 1.748446

track leg statistics for the terminal legs 13 and 14

leg 13 length = 5.341702
leg 14 length = 3.676944

In Table I, the leg course phi (ϕ) which is for a track whose initial point and end point lie on an east west line and the track leg angle theta (θ) are rounded to integer degrees. Other than the requirement of consistency, the units for length are arbitrary. The coordinates X and Y in the track data printout refer to the first rectangular coordinate system. In this system, the origin is at the midpoint of the line joining the initial point and the end point of the track, the direction of the positive x-axis is east and the direction of the positive y-axis is north.

Figure 4 is a composite of 1000 tracks that were generated by an extension of the program listed in Appendix 2. Except for the track length which is 20 units, the conditions for the generation of the tracks are identical to the conditions for the generation of the track that is shown in Figure 3. Table II which follows Figure 4 lists some data associated with the tracks.

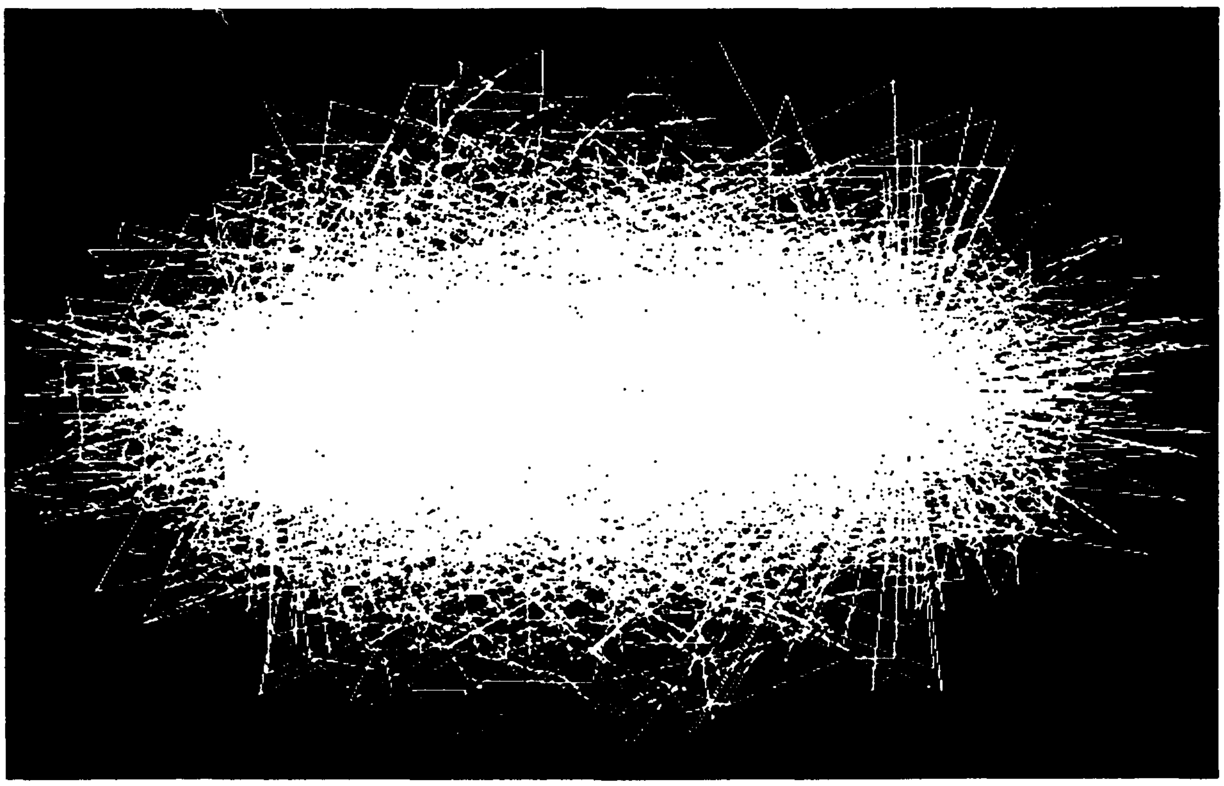

Figure 4. Tracks generated by an extension of the program that is listed in Appendix 2. The bounding ellipse for their first legs forms an envelope around the tracks. Track data is listed in Table II.

Table II Some data associated with the tracks shown in Figure 4. In the table, the next to last leg is the first terminal leg and the last leg is the second terminal leg.

stationary leg bearing and range distributions
distance between the end points 10
track length = 20
delta = 3
leg length distribution index = 1
maximum number of legs allowed = 30
program random number generator
random number seed = 123

number of tracks = 1000
number of completed tracks = 1000
number of tracks with more than two legs = 6

average nonterminal leg track length = 12.26772
average nonterminal leg length = 1.66974
average maximum nonterminal leg length = 4.838235
average first terminal leg length = 2.500977
average second terminal leg length = 5.304918

maximum track maximum nonterminal leg length = 13.00095
maximum first terminal leg length = 14.96789
maximum second terminal leg length = 14.56736

probability first terminal leg be greater than maximum nonterminal leg = .206
probability second terminal leg greater than maximum nonterminal leg = .489
probability second terminal leg greater than first terminal leg = .713

Appendix 1. Some Mathematical Relationships

In the simulation program, values of the random variables Θ_k and R_k^* are generated by an inverse transform method. In particular, for Θ_k, it is set equal to its inverse cumulative distribution function with the argument a random variable P with a uniform distribution on the interval 0 to 1. This generates a one to one correspondence between the values of Θ_k and the values of P through the inverse cumulative distribution function. For a dynamic process, for each leg except the last, the cumulative distribution function for Θ_k is defined by:

$$F_{\Theta_k}(\theta_k) = \int_{-\pi/2}^{\theta_k} \frac{1}{2\pi} \frac{b_k}{a_k - c_k \sin\theta} d\theta \ .$$

And, for $-\pi/2 \le \theta_k < \pi/2$:

$$F_{\Theta_k}(\theta_k) = \frac{1}{\pi}\{ \tan^{-1}\left[\frac{a_k \tan(\theta_k/2) - c_k}{b_k}\right] - \tan^{-1}\left[\frac{-(a_k + c_k)}{b_k}\right] \} \ .$$

And, for $\pi/2 \le \theta_k < 3\pi/2$:

$$F_{\Theta_k}(\theta_k) = 1 - F_{\Theta_k}(\pi - \theta_k) \ .$$

For the distribution index $n = 1$, R_k^* has an exponential distribution and values for R_k^* are simulated using its cumulative distribution function with the inverse transform method in the same way that they are for Θ_k. For $n = 1$ and a dynamic process, for each leg except the last, the cumulative distribution function for R_k^* given $\Theta_k = \theta_k$ is defined by:

$$F_{R_k^*|\Theta_k}(r_k^*|\theta_k) = \int_0^{r_k^*} \frac{1}{\delta(\theta_k)} e^{-\frac{r}{\delta(\theta_k)}} dr$$

$$= 1 - e^{-\frac{r_k^*}{\delta(\theta_k)}} .$$

With $n > 1$, R_k^* is equal to the sum of n independent exponentially distributed random variables each with mean $\delta(\theta_k)/n$. Consequently, for $n > 1$ a value for R_k^* can be generated by first generating a value for each of the n exponentially distributed random variables and then summing the resulting values.

By symmetry, given the termination criterion has not been satisfied, the expected value of the random variable Θ_k is $\pi/2$. The determination of the expected value of the random variable R_k for this case is not so easy. Without the length constraint that is introduced by the bounding ellipse, $R_k = R_k^*$ and, for the distribution index $n = 1$, its expected value is given by:

$$E(R_k^*) = 2\int_{-\pi/2}^{\pi/2} E(R_k^*|\Theta_k=\theta_k)\, f_{\Theta_k}(\theta_k)\, d\theta_k$$

$$= 2\int_{-\pi/2}^{\pi/2} \frac{\delta}{\pi} \frac{b_k\,(a_k - c_k)}{(a_k - c_k \sin\theta)^2}\, d\theta_k$$

$$= 2\,\frac{\delta}{\pi}\, \frac{a_k}{a_k + c_k} \{\tan^{-1}\left(\frac{b_k}{a_k + b_k}\right) + \tan^{-1}\left(\frac{a_k + b_k}{b_k}\right)\}$$

$$= \delta\, \frac{a_k}{a_k + c_k} .$$

Consequently, for $n > 1$, since R_k^* is equal to the sum of n independent exponential random variables each with mean $\delta(\theta_k)/n$, the mean of R_k^* generated by the simulation is given by:

$$\mu_{R_k^*} = \delta \frac{a_k}{a_k + c_k} .$$

And its standard deviation is given by:

$$\sigma_{R_k^*} = (\delta / \sqrt{n}) \frac{a_k}{a_k + c_k} .$$

For a static process, a replaces a_k, b replaces b_k and c replaces c_k in the above expressions.

If the track termination criterion is satisfied on the k^{th} leg, then the track will terminate on the $k+1^{st}$ leg. For a dynamic process with the distribution index $n = 1$, the probability on the k^{th} leg that a track will terminate on the $k+1^{st}$ leg is:

$$\begin{aligned} P_{k+1} &= 2\int_{-\pi/2}^{\pi/2} P[R_k^* > r_k(\theta_k)]\, f_{\theta_k}(\theta_k)\, d\theta_k \\ &= 2\int_{-\pi/2}^{\pi/2} e^{-\frac{r_k(\theta_k)}{\delta(\theta_k)}} f_{\theta_k}(\theta_k)\, d\theta_k \\ &= e^{-\frac{a_k + c_k}{\delta}} . \end{aligned}$$

For $k = 1$, this also applies for a static process.

Appendix 2. The Simulation Program Listing

The program code that is listed below generates random tracks using the BASIC random number generator. An auxiliary random number generator can be added through Lines 200, 210 and 220. These lines provide for the generator setup code in a subroutine starting at Line 1710 and for the generator code in a subroutine starting at Line 1820. The choice of this line number was governed by code for a Generalized Feedback Shift Register (GFSR) pseudorandom number generator that was investigated during the simulation program's development. The code was adapted for use in a 32 bit machine from code that is listed in Reference 3. If an auxiliary generator will not be added, Lines 200, 210 and 220 can be removed.

The program requires a computer monitor with screen mode 12 capability. To change this requirement, the statement SCREEN 12 on Line 340 must be changed. For example, for a monitor with CGA capability, it could be replaced by SCREEN 2.

```
10 REM RTRACK.BAS, a random track generating program
20 CLS : DEFINT M-N
30 PI = 4 * ATN(1): R$ = "standard": FLAG0 = 0: FLAG1 = 0
40 A$ = " ": PRINT : INPUT "generate track data or print a track data file (g/p)"; A$
50 IF A$ = "P" OR A$ = "p" THEN 1400
60 IF A$ = "G" OR A$ = "g" THEN 70 ELSE 40
70 A$ = " ": : PRINT : INPUT "stationary or dynamic leg bearing and range distributions (s/d)"; A$
80 IF A$ = "S" OR A$ = "s" THEN D$ = "stationary": GOTO 110
90 IF A$ = "D" OR A$ = "d" THEN FLAG1 = 1 ELSE 70
100 D$ = "dynamic"
110 PRINT : INPUT "distance between the end points"; RANGE
120 PRINT : INPUT "track length"; LEGSUM: IF LEGSUM <= RANGE THEN 120
130 IF RANGE > LEGSUM / 3 THEN D = 2 * RANGE / 3 ELSE D = LEGSUM / 3
140 PRINT : INPUT "delta"; DEL
150 IF DEL <= 0 THEN 140
160 A$ = " ": PRINT : INPUT "leg length distribution index"; N
170 IF N < 1 THEN 160
```

```
180 PRINT : INPUT "maximum number of legs allowed"; MNL
190 IF MNL < 2 THEN 180
200 A$ = " ": PRINT : INPUT "standard or auxiliary random number generator (s/a)"; A$
210 IF A$ = "A" OR A$ = "a" THEN GOSUB 1730: GOTO 280
220 IF A$ = "S" OR A$ = "s" THEN GOTO 230 ELSE 200
230 A$ = " ": PRINT : INPUT "supply a random number seed (y/n)"; A$
240 IF A$ = "N" OR A$ = "n" THEN 280
250 IF A$ = "Y" OR A$ = "y" THEN FLAG0 = 1 ELSE 230
260 PRINT : INPUT "random number seed"; RNS
270 IF RNS < -32768 OR RNS > 32767 THEN 260 ELSE RANDOMIZE RNS
280 DIM F(MNL), LEG(MNL), T(MNL), X(MNL), Y(MNL)
290 X(0) = -RANGE / 2: Y(0) = 0: F(0) = 0: LEG(0) = 0: T(0) = 0
300 ALEG = 0: LEGMAX = 0
310 FLAG2 = 0
320 A0 = LEGSUM / 2: C0 = RANGE / 2
330 A = A0: C = C0: S = 0
340 SCREEN 12: WINDOW (-D, -D)-(D, D)
350 CIRCLE (X(0), Y(0)), .01 * D: CIRCLE (-X(0), Y(0)), .01 * D
360 FOR I = 1 TO MNL
370 K0 = SQR((A0 + C0) / (A0 - C0))
380 K1 = SQR(1 - C0 * C0 / A0 / A0)
390 GOSUB 1710
400 ON ERROR GOTO 410: GOTO 420
410 RESUME 390
420 FLAG3 = 0
430 IF RAND <= .5 THEN 450
440 RAND = RAND - .5: FLAG3 = 1
450 T = 2 * ATN(K1 * TAN(PI * RAND - ATN(K0)) + C0 / A0)
460 ON ERROR GOTO 0
470 IF FLAG3 = 1 THEN T = PI - T
480 P = (A0 - C0) / (A0 - C0 * SIN(T))
490 GOSUB 1630
500 ON ERROR GOTO 0
510 F = T + S
520 LEGA = (A * A - C * C) / (A - C * SIN(T))
530 IF LEG > LEGA THEN LEG = LEGA ELSE 550
540 FLAG2 = 1
550 GOSUB 1550
560 LINE (X(I), Y(I))-(X(I - 1), Y(I - 1))
570 A = A1: C = C1: S = AN - PI / 2: LEG(I) = LEG
580 IF FLAG1 = 1 THEN A0 = A1: C0 = C1
590 H = T: GOSUB 1520
600 T(I) = H
610 H = F: GOSUB 1520
620 F(I) = H
630 IF FLAG2 = 1 THEN 680
640 ALEG = ALEG + LEG
650 IF LEGMAX < LEG THEN LEGMAX = LEG
660 NEXT I
670 PRINT : PRINT "maximum number of legs exceeded and track terminated"
680 LINE (RANGE / 2, 0)-(X(I), Y(I))
690 NL = I + 1: LEG(NL) = 2 * C: T(NL) = 90: H = PI / 2 + S
```

```
700 GOSUB 1520: F(NL) = H: X(NL) = RANGE / 2
710 IF I < 2 THEN ALEG = 0: GOTO 730
720 ALEG = ALEG / (NL - 2)
730 A$ = " ": PRINT : INPUT "generate an additional track (y/n)"; A$
740 IF A$ = "N" OR A$ = "n" THEN 770
750 IF A$ = "Y" OR A$ = "y" THEN 760 ELSE 730
760 SCREEN 0: ERASE F, LEG, T, X, Y: GOTO 280
770 A$ = " ": PRINT : INPUT "quit the program (y/n)"; A$
780 IF A$ = "N" OR A$ = "n" THEN 800
790 IF A$ = "Y" OR A$ = "y" THEN 1510 ELSE 770
800 A$ = " ": PRINT : INPUT "print the track data (y/n)"; A$
810 IF A$ = "N" OR A$ = "n" THEN 1050
820 IF A$ = "Y" OR A$ = "y" THEN 830 ELSE 800
830 LPRINT D$ + " leg bearing and range distributions"
840 LPRINT "distance between the end points ="; RANGE
850 LPRINT "track length ="; LEGSUM
860 LPRINT "delta ="; DEL
870 LPRINT "leg length distribution index ="; N
880 LPRINT "maximum number of legs allowed ="; MNL
890 LPRINT "number of legs ="; NL
900 LPRINT R$ + " random number generator"
910 IF FLAG0 = 0 THEN 930
920 LPRINT "random number seed ="; RNS
930 LPRINT : LPRINT : LPRINT " i"; TAB(9); "phi"; TAB(18); "leg length"; TAB(32); "theta";
TAB(42); "X"; TAB(58); "Y"
940 LPRINT
950 FOR I = 0 TO NL
960 LPRINT I; TAB(8); F(I); TAB(17); LEG(I); TAB(31); T(I); TAB(41); X(I); TAB(57); Y(I)
970 NEXT I
980 LPRINT : LPRINT : L$ = STR$(NL - 2): M$ = STR$(NL - 1): N$ = STR$(NL)
990 LPRINT "track leg statistics for the nonterminal legs 1 through" + L$
1000 LPRINT : LPRINT "maximum nonterminal leg length ="; LEGMAX
1010 LPRINT "average nonterminal leg length  ="; ALEG
1020 LPRINT : LPRINT "track leg statistics for the terminal legs" + M$ + " and" + N$
1030 LPRINT : LPRINT "leg" + M$ + " length ="; LEG(NL - 1)
1040 LPRINT "leg" + N$ + " length ="; LEG(NL)
1050 A$ = " ": PRINT : INPUT "print the track data to a file (y/n)"; A$
1060 IF A$ = "N" OR A$ = "n" THEN 1360
1070 IF A$ = "Y" OR A$ = "y" THEN 1080 ELSE 1050
1080 PRINT : INPUT "input the data file name"; F$
1090 ON ERROR GOTO 1100: GOTO 1110
1100 RESUME 1080
1110 OPEN "O", #1, F$
1120 PRINT #1, D$ + " leg bearing and range distributions"
1130 PRINT #1, "distance between the end points = "; RANGE
1140 PRINT #1, "track length = "; LEGSUM
1150 PRINT #1, "delta = "; DEL
1160 PRINT #1, "leg length distribution index ="; N
1170 PRINT #1, "maximum number of legs allowed ="; MNL
1180 PRINT #1, "number of legs ="; NL
1190 PRINT #1, R$ + " random number generator"
1200 IF FLAG0 = 0 THEN 1220
```

```
1210 PRINT #1, "random number seed ="; RNS
1220 PRINT #1, : PRINT #1, : PRINT #1, " i"; TAB(9); "phi"; TAB(18); "leg length"; TAB(32); "theta";
TAB(42); "X"; TAB(58); "Y"
1230 PRINT #1,
1240 FOR I = 0 TO NL
1250 PRINT #1, I; TAB(8); F(I); TAB(17); LEG(I); TAB(31); T(I); TAB(41); X(I); TAB(57); Y(I)
1260 NEXT I
1270 PRINT #1, : PRINT #1, : L$ = STR$(NL - 2): M$ = STR$(NL - 1): N$ = STR$(NL)
1280 PRINT #1, "track leg statistics for the nonterminal legs 1 through" + L$
1290 PRINT #1, : PRINT #1, "maximum nonterminal leg length ="; LEGMAX
1300 PRINT #1, "average nonterminal leg length ="; ALEG
1310 PRINT #1, : PRINT #1, "track leg statistics for the terminal legs" + M$ + " and" + N$
1320 PRINT #1, : PRINT #1, "leg" + M$ + " length ="; LEG(NL - 1)
1330 PRINT #1, "leg" + N$ + " length ="; LEG(NL)
1340 CLOSE #1
1350 ON ERROR GOTO 0
1360 A$ = " ": PRINT : INPUT "continue the program (y/n)"; A$
1370 IF A$ = "N" OR A$ = "n" THEN 1510
1380 IF A$ = "Y" OR A$ = "y" THEN 1390 ELSE 1360
1390 RUN 10
1400 PRINT : INPUT "input the data file name"; F$
1410 ON ERROR GOTO 1420: GOTO 1430
1420 RESUME 1400
1430 OPEN "I", #1, F$
1440 DO UNTIL EOF(1)
1450 LINE INPUT #1, LINEBUFFER$
1460 LPRINT LINEBUFFER$
1470 LOOP
1480 CLOSE #1
1490 ON ERROR GOTO 0
1500 GOTO 1360
1510 END
1520 H = H * 180 / PI: H = H - 360 * INT(H / 360): IF H < 0 THEN H = H + 360
1530 H1 = INT(H): IF H - H1 >= .5 AND H1 < 360 THEN H = H1 + 1 ELSE H = H1
1540 RETURN
1550 X(I) = X(I - 1) + LEG * SIN(F)
1560 Y(I) = Y(I - 1) + LEG * COS(F)
1570 X = RANGE / 2 - X(I): Y = -Y(I)
1580 R = SQR(X * X + Y * Y): C1 = R / 2: A1 = A - LEG / 2: IF R = 0 THEN AN = 0: GOTO 1620
1590 IF ABS(X / R) = 1 THEN AM = PI / 2 * SGN(X) ELSE AM = ATN(X / R / SQR(1 - X * X
/ R / R))
1600 IF ABS(Y / R) = 1 THEN AN = PI / 2 * (1 - SGN(Y)) ELSE AN = PI / 2 - ATN(Y / R / SQR(1
- Y * Y / R / R))
1610 IF AM < 0 THEN AN = 2 * PI - AN
1620 RETURN
1630 LEG = 0
1640 FOR L = 1 TO N
1650 GOSUB 1710
1660 ON ERROR GOTO 1670: GOTO 1680
1670 RESUME 1650
1680 LEG = -DEL * P * LOG(1 - RAND) / N + LEG
1690 NEXT L
```

```
1700 RETURN
1710 IF R$ = "standard" THEN RAND = RND ELSE GOSUB 1820: REM auxiliary generator
subroutine branch
1720 RETURN
1730 R$ = "auxiliary": REM the first line of an auxiliary generator setup subroutine
```

References

1. Forrest, R. N., "A Random Tour Process of Known Length between Known End Points," NPS71-83-001, Naval Postgraduate School, Monterey, CA 93943, January 1983.

2. Washburn, Alan R. 1968, "Probability Density of a Moving Particle," Opns. Res., 17, 861-871.

3. Macomber, James H. and White, Charles S. 1990, "An n-Dimensional Uniform Random Number Generator Suitable for IBM-Compatible Microcomputers," Interfaces, 20, 49-59.

Initial Distribution List

	Copies
Director of Research Administration (Code 012) Naval Postgraduate School Monterey, CA 93943-5000	1
Defense Technical Information Center Cameron Station Alexandria, VA 22314	2
Library (Code 0142) Naval Postgraduate School Monterey, CA 93943-5000	2
Chief of Naval Operations OP-714C Washington, D. C. 20350-2000	1
Chief of Naval Operations OP-21T1 Washington, D. C. 20350-2000	1
Commander Submarine Development Squadron 12 Naval Submarine Base, New London Groton, CT 06349	1
Commander Surface Warfare Development Group Naval Amphibious Base, Little Creek Norfolk, VA 23521	1
Naval Air Development Center Attn: W. Leyland, Code 3C1 Johnsville, PA 18974	1
Naval Surface Weapons Center White Oak Silver Spring, MD 20910	1
Naval Underwater Systems Center Newport, RI 02840	1
Naval Underwater Systems Center New London, CT 06320	1
Naval Technical Intelligence Center Attn: Library and G. Batts, Code DA 23 4301 Suitland Road Washington, D. C. 20390	2

Naval Research Laboratory Washington, D. C. 20375	1
Naval Ocean Systems Center San Diego, CA 92132	1
Director, Wargaming Department Naval War College Attn: Code 3314B and Code 33T Newport, RI 02841	2
Center for Naval Analysis 4401 Ford Avenue P. O. Box 16268 Alexandria, VA 22302-0268	1
Applied Physics Laboratory Attn: R. N. McDonough Johns Hopkins Road Laurel, MD 20707	1
ORI Inc. Attn: P. L. Halowell 1400 Spring St. Silver Spring, MD 20910	1
R and D Associates Attn: A. G. Cicolani 1401 Wilson Blvd. Arlington, VA 22209	1
D. H. Wagner, Associates Station Square One Paoli, PA 19301	1
Metron, Inc. 1481 Chain Bridge Road McLean, VA 22101	1
School of Business Administration Attn: James H. Macomber University of Tennessee at Chattanooga Chattanooga, Tennessee 37403	1
Naval Postgraduate School Monterey, CA 93943-5000 Code OR/Fo	25